Theor Chim Acta (1996) 93: 35–48

Theoretica Chimica Acta

Graphical representation of a new algorithm for nonorthogonal *ab initio* valence bond calculations

Jiabo Li*

Department of Theoretical Chemistry, University of Erlangen-Nürnberg, Egerlandstrasse 3, D-91058 Erlangen, Germany

Received June 2, 1995/Accepted June 2, 1995

Abstract. A pictorial representation of the algorithm using successive expansion method for the nonorthogonal VB calculations is given. With the help of this representation and the graph analysis, the efficiency of this algorithm is improved and the $N!$ problem is reduced by a factor of about $(N!)^{1/2}$. An *ab initio* VB program for valence bond self-consistent-field (VBSCF) calculations has been implemented based on this algorithm. Some VBSCF calculations have been performed for systems of up to 14 electrons. The statistics of the CPU time of the calculations indicate that this new group-theoretical approach is quite practical.

Key words: VB method – $N!$ problem – Group-theoretical approach – VBSCF calculations

1 Introduction

The MO-based methods are capable of providing reliable description of the electronic structures of molecular systems. However, it is difficult to interpret the highly accurate results with the qualitative terms such as valence [1], electron pairs and resonance [2], while these well-established concepts are particularly useful to rationalize chemical phenomena. Modern valence bond method provides a conceptual tool for the quantitative interpretation of chemical phenomena. A large number of successful applications of nonorthogonal VB approach to a wide range area of chemistry, including chemical bonding [4–6], reactions [7–9] and drugs [10], have been seen in recent years. A significant role in the success of the VB calculations is played by using nonorthogonal optimized orbitals for the construction of the VB wave functions. However, the major drawback of the VB method is the $N!$ problem due to the use of nonorthogonal orbitals. Owing to many years' efforts, considerable progress has been achieved for the efficient implementations of the nonorthogonal VB method based on the Slater-determinant expansion [11–18]. Another method is the group-theoretical approach [19–26] based on the

* *Permanent address*: Chemistry Department of Xiamen University, Fujian, 361005 Xiamen, P.R. China

spin-free formalism [27, 28]. Although the mathematics of the spin-free VB formalism has been extensively investigated, most group theoretical approaches do not lead to a really practical algorithm. Some new technical developments in the group-theoretical approach have been made recently [24, 25]. For a system of spin $S = 0$, the VB calculation is reduced to the summation of $(N-1)!!$ permanents of order n $(= N/2)$. By using the successive Laplace expansion method, the $n!$ problem of a permanent is reduced to an $n \times 2^{n-1}$ problem, thus the $N!$ problem is reduced to an $(2n-1)!!n \times 2^{n-1}$ problem [24]. To achieve a further reduction of the computational effort, a successive expansion procedure was introduced for the systematic collection of a large number of permanents [25]. In this scheme, the so-called "contracted cofactors" are used as the intermediate quantities. As the evaluation of individual permanents is avoided in this scheme, significant reduction of the computational effort can be achieved. To find a really powerful algorithm, the combination of computational techniques and fundamental-theoretical research is quite necessary. The introduction of graphical techniques, such as graphical unitary group approach (GUGA) [29, 30] and the symmetric group-graphical approach (SGGA) [31] increases the effectiveness of the group-theoretical approach to a great extent [26b]. Some sophisticated VB algorithms can also be represented in a much clearer way in graphical language [14, 18]. It should be mentioned here that the "contracted cofactors" introduced in Ref. [25] are actually the overlaps of VB wave functions of various number of electrons, and the VB wave functions can be characterized by the pairing patterns of various number of orbitals. In this work, the successive expansion method is further explored and the procedure can be characterized by the successive decomposition of the pairing patterns. With the help of this new representation and graph analysis, a further improved algorithm is proposed. This method has been implemented in a VB program for multiple-VB-structure calculations and VBSCF calculations. Some primary VBSCF calculations have been performed for CH_4, C_2H_2, C_2H_4 and C_2H_6. The practice shows that the CPU time increases in a reasonable manner with N.

2 Graphical representation of the successive expansion method

Spin-free form of VB wave functions

The VB wave functions of N electrons can be expressed as the following:

$$\Psi_{\mathrm{VB}} = e_{11}^{[\lambda]}\Omega, \tag{1}$$

where $e_{11}^{[\lambda]}$ is the Wigner operator written as

$$e_{11}^{[\lambda]} = \left(\frac{f_\lambda}{N!}\right)^{1/2} \sum_{P \in S_N} D_{11}^{[\lambda]}(P)P. \tag{2}$$

$D_{11}^{[\lambda]}(P)$ are matrix elements of the irrep $[\lambda] = [2^{1/2N-S}, 1^{2S}]$, f_λ is the dimension of the irrep, and Ω is a product of N orbitals. The overlap and Hamiltonian matrix elements of VB wave functions can be expressed as

$$\langle \Psi_{\mathrm{VB}} | \Psi'_{\mathrm{VB}} \rangle = \sum_{P \in S_N} D_{11}^{[\lambda]}(P) \langle \Omega | P | \Omega' \rangle, \tag{3}$$

$$\langle \Psi_{\mathrm{VB}} | H | \Psi'_{\mathrm{VB}} \rangle = \sum_{P \in S_N} D_{11}^{[\lambda]}(P) \langle \Omega | PH | \Omega' \rangle, \tag{4}$$

In the following discussion, we restrict our consideration to the case $S = 0$. To make the notations more concise, the superscript $[\lambda]$ is omitted hereafter. Let us consider how to evaluate the D_{11}-values in a graphical way. Suppose we have a permutation P as below:

$$P = \begin{pmatrix} 1 & 2 & \cdots & N \\ p_1 & p_2 & \cdots & p_N \end{pmatrix}, \tag{5}$$

we can represent it in a two-column-box form as follows: the numbers $p_1, p_2, \ldots, p_N$ are filled in these boxes from the left to the right and from the top to the bottom:

$$P = \begin{array}{|c|c|} \hline p_1 & p_2 \\ \hline \vdots & \vdots \\ \hline p_{2k-1} & p_{2k} \\ \hline \vdots & \vdots \\ \hline p_{N-1} & p_N \\ \hline \end{array} \tag{6}$$

To evaluate $D_{11}(P)$, first we draw lines between every two elements in the following way [24, 26]: 1–2, 3–4, $\ldots$, $(2k-1)-2k$, $\ldots$ $(N-1)$–N, then draw lines between every two elements of the pairs shown above, i.e. p_1–p_2, $\ldots$, p_{2k-1}–p_{2k}, $\cdots$ p_{N-1}–p_N; therefore, we get a graph of N elements linked by N lines. These lines form some topological loops. Suppose the number of loops is L, then we have

$$D_{11}(P) = (-\tfrac{1}{2})^{N/2-L}. \tag{7}$$

From the above discussion, we can see easily that if the pairs in the two-column boxes are preserved, then neither the interchange of the two elements of a pair nor the interchange of the pairs changes the D_{11}-values of the permutations. Thus for each pairing pattern, we can generate a set of $2^{N/2}(N/2)!$ permutations which have the same D_{11}-value. One can generate all the permutations of S_N by choosing all possible pairing patterns. For the evaluation of a VB overlap matrix element, it turns out that the summation over each set of permutations corresponds to a permanent of a matrix of order $N/2$. Thus, we have

$$\langle \Psi_{\mathrm{VB}} | \Psi'_{\mathrm{VB}} \rangle = \sum_i D_{11}^{[\lambda]}(G_i^N)\mathrm{per}(G_i^N), \tag{8}$$

where G_i^N is a pairing pattern of N orbitals, and per (G_i^N) is the corresponding permanent of order $N/2$. The first summation runs over all $(N-1)!!$ pairing patterns. Although the $n!$ problem of a permanent can be significantly reduced [24, 32], a direct use of this expression is not very practical. As shown in a previous paper, the successive expansion method is much more efficient than the direct summation. In the following discussion, this procedure is represented in a simple way.

Graphical notations

It is convenient to introduce some notations in the following discussions. Suppose G and G' are two patterns of n pairs of orbitals as follows:

$$G = \begin{Bmatrix} \phi_1 & - & \phi_2 \\ \phi_3 & - & \phi_4 \\ & \vdots & \\ \phi_{2n-1} & - & \phi_{2n} \end{Bmatrix}, \qquad G' = \begin{Bmatrix} \phi'_1 & - & \phi'_2 \\ \phi'_3 & - & \phi'_4 \\ & \vdots & \\ \phi'_{2n-1} & - & \phi'_{2n} \end{Bmatrix} \tag{9}$$

we use $|G\rangle$ and $|G'\rangle$ to denote the corresponding VB wave functions of $2n$ electrons

$$|G\rangle \equiv |\Psi\rangle = e_{11}^{[\lambda]}\Omega$$

$$= e_{11}^{[\lambda]}\phi_1(1)\ \phi_2\ (2)\cdots\ \phi_{2n}(2n), \tag{10}$$

$$|G'\rangle \equiv |\Psi'\rangle = e_{11}^{[\lambda]}\Omega'$$

$$= e_{11}^{[\lambda]}\phi'_1(1)\ \phi'_2\ (2)\cdots\ \phi'_{2n}(2n), \tag{11}$$

These wave functions describe the chemical bonding of the orbital pairs as shown in their corresponding pairing patterns. We define the overlap of two pairing patterns as the overlap of the two corresponding VB wave functions, i.e.

$$\langle G|G'\rangle \equiv \langle\Psi|\Psi'\rangle = \sum_{P\in S_{2n}} D_{11}(P)\langle\Omega|P|\Omega'\rangle \tag{12}$$

$$= \sum_i D_{11}(G_i^N)\mathrm{per}(G_i^N).$$

One can define in a similar way the overlap of two subpatterns of m ($m < n$) pairs as the overlap of the corresponding VB wave functions of $2m$ electrons.

Expansion of $\langle G|G'\rangle$ using the overlaps of subpatterns

With the overlaps of subpatterns, one can express the VB overlap matrix elements in a way which is quite similar to the Laplace expansion of a determinant:

$$\langle G|G'\rangle = \sum_{G_A} d_{11}(G_A) \sum_{i=1}^{n} A_i\langle G^A|G'^i\rangle, \tag{13}$$

where

$$G_A = \phi_1 - \phi_j \quad (j = 2,3,\ \ldots\ ,2n)$$

and

$$d_{11}(G_A) = \begin{cases} 1 & \text{if } j = 2, \\ -\dfrac{1}{2} & \text{otherwise.} \end{cases}$$

G^A is the subpattern of G without the first pair and the orbital ϕ_j in the subpattern is replaced by ϕ_2. For example,

$$G_A = \phi_1 - \phi_{2m-1}, \quad \text{then } G_A = \begin{cases} \phi_3 & - & \phi_4 \\ & \vdots & \\ \phi_2 & - & \phi_{2m} \\ & \vdots & \\ \phi_{2n-1} & - & \phi_{2n}. \end{cases} \tag{14}$$

G'^i is the subpattern of G' without the ith pair. A_i can be expressed as

$$\begin{aligned} A_i &= \langle G_A | G'_i \rangle \quad (G'_i = \phi'_{2i-1} - \phi'_{2i}) \\ &= \langle \phi_1 | \phi'_{2i-1} \rangle \langle \phi_j | \phi'_{2i} \rangle + \langle \phi_j | \phi'_{2i-1} \rangle \langle \phi_1 | \phi'_{2i} \rangle, \end{aligned} \tag{15}$$

and $|G^A\rangle$ and $|G'^i\rangle$ denote the VB wave functions of $(2n-2)$ electrons corresponding to the two subpatterns G^A and G'^i, respectively. Obviously, if all the quantities $\langle G^A | G'^i \rangle$ are available, it requires only $n(2n-1)$ multiplications to compute the quantity $\langle G | G' \rangle$. The mathematical proof of the above expressions is trivial. One can find all necessary details in Refs. [24, 25] for the proof.

Successive expansion of the overlaps of the subpatterns

From the above discussion, we know that it will be quite easy to evaluate $\langle G | G' \rangle$ if all the overlap elements $\langle G^A | G'^i \rangle$ are available. Therefore, the next problem is how to evaluate all these quantities in an efficient way. Obviously, the same expansion method can be applied to the evaluation of these overlaps and the quantities of lower subpatterns successively. Equations (13) and (14) show that from G one can generate $2n-1$ subpatterns G^A, and for each G^A there are n overlap elements $\langle G^A | G'^i \rangle$. From the same reasoning, we know that from each G^A there are n overlap elements $\langle G^A | G'^i \rangle$. From the same reasoning, we know that from each G^A one obtains $2n-3$ lower subpatterns G^{AB}, and for each G^{AB} there are $n(n-1)/2$ overlaps $\langle G^{AB} | G'^{ij} \rangle$ $(i < j = 1,2, \ldots, n)$. That means we can compute $\langle G^A | G'^i \rangle$ by using the overlaps of the lower subpatterns $\langle G^{AB} | G'^{ij} \rangle$. This procedure can be continued, and can be represented schematically as follows:

patterns	overlaps
G	$\langle G \mid G' \rangle$
$\downarrow$	$\uparrow$
$N_G(n-1)$ G^A	$\langle G^A \mid G'^i \rangle$ $(i = 1,2, \ldots, n)$
$\downarrow$	$\uparrow$
$N_G(n-2)$ G^{AB}	$\langle G^{AB} \mid G'^{ij} \rangle$ $(i < j = 1,2, \ldots, n)$
$\vdots$	$\vdots$
$N_G(m)$ $G^{AB\ldots D}$	$\langle G^{AB\ldots D} \mid G'^{ij\ldots l} \rangle$ $(i < j < \cdots < l = 1,2, \ldots, n)$
$\downarrow$	$\uparrow$

where $N_G(m)$ denotes the number of unique subpatterns of m pairs. Obviously, $N_G(n) = 1$ and $N_G(n-1) = n$. Table 1 gives these numbers. Therefore, to compute

Table 1. Number of unique subpatterns for the evaluation of one overlap matrix element of VB wave functions of N electrons

N	Number of pairs in the subpatterns							
	2	3	4	5	6	7	8	9
6	5							
8	15	7						
10	51	28	9					
12	102	148	45	11				
14	222	370	325	66	13			
16	370	1390	975	606	91	15		
18	650	2780	5415	2121	1015	120	17	
20	975	7200	12635	15771	4060	1576	153	19

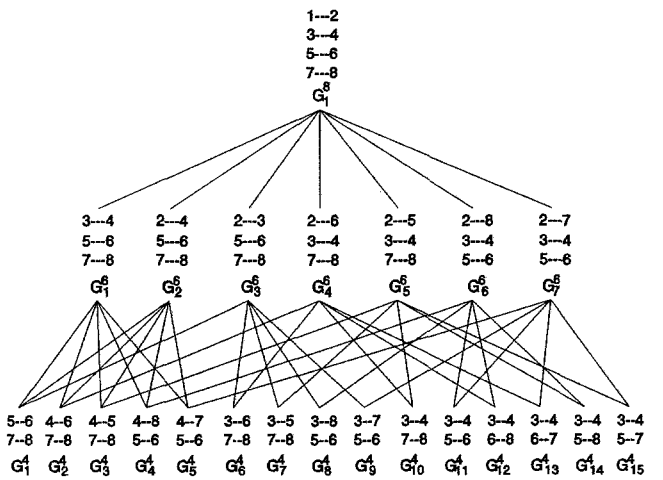

Fig. 1. The decomposition of a pattern of 8 elements with the lexical ordering (see Sect. 3). The lines indicate from which higher subpattern a lower subpattern is generated

the overlap matrix element $\langle G | G' \rangle$, one first performs the pairing-pattern-decomposition and obtains a tree of patterns of various numbers of pairs. A simple example of a pattern-decomposition tree is shown in Fig. 1.

In principle, this pattern-decomposition analysis is required only once for a certain number of electrons and spin. In actual computation, one starts from the computation of the overlaps of the lowest subpatterns at the bottom of the tree. The quantities corresponding to the lower subpatterns are repeatedly used for the computation of the higher ones. Similar recursive strategy for the evaluation of density matrices of various orders was discussed by Gerratt [33] and applied by Pyper and Gerratt [34]. Finally, one obtains the VB overlap matrix element at the top of the tree. The total number of multiplications required in the whole procedure can be given as

$$M = \sum_{m=2}^{n} N_G(m) \frac{n!\,(2m-1)}{(n-m)!\,(m-1)!}. \tag{16}$$

One should point out that the pattern-decomposition is not unique. By choosing a different way of decomposition, one obtains a different number of M. One essential problem is how to find a way which gives a small M. This will be discussed in Sect. 4 in more detail.

Matrix elements of the Hamiltonian

The above strategy can also be applied to the evaluation of the Hamiltonian matrix elements. It is convenient to define a function $P_A(m)$ as follows:

$$\begin{aligned} P_A(2m-1) &= 2m \\ P_A(2m) &= 2m-1 \end{aligned} \qquad (m = 1,2, \ldots). \tag{17}$$

One can partition a Hamiltonian matrix element into two parts

$$\langle G|H|G'\rangle = H_1 + H_2. \tag{18}$$

H_1 is the contribution of the one-electron operators and partially of the two-electron operators:

$$H_1 = \sum_{a<b}^{2n} d_{11}(G_A) \sum_{i=1}^{n} F_i^A \langle G^A | G'^i\rangle, \tag{19}$$

where

$$G_A = \phi_a - \phi_b \quad \text{and} \quad d_{11}(G_A) = \begin{cases} 1 & \text{if } P_A(a) = b, \\ -\frac{1}{2} & \text{otherwise.} \end{cases}$$

G^A is a subpattern of G without the elements ϕ_a and ϕ_b, and G'^i is a subpattern of G' without the ith pair. If $P_A(a) \neq b$, then one pair in G^A is

$$\phi_{P_A(a)} - \phi_{P_A(b)}$$

F_i^A can be evaluated as

$$\begin{aligned} F_i^A = {} & \langle \phi_{a_1}|f(1)|\phi'_{2i-1}\rangle \langle \phi_{a_2}|\phi'_{2i}\rangle + \langle \phi_{a_2}|f(1)|\phi'_{2i-1}\rangle \langle \phi_{a_1}|\phi'_{2i}\rangle \\ & + \langle \phi_{a_1}|\phi'_{2i-1}\rangle \langle \phi_{a_2}|f(1)|\phi'_{2i}\rangle + \langle \phi_{a_2}|\phi'_{2i-1}\rangle \langle \phi_{a_1}|f(1)|\phi'_{2i}\rangle \\ & + \langle \phi_{a_1}\phi_{a_2}|g(1,2)|\phi'_{2i-1}\phi'_{2i}\rangle + \langle \phi_{a_2}\phi_{a_1}|g(1,2)|\phi'_{2i-1}\phi'_{2i}\rangle. \end{aligned} \tag{20}$$

H_2 is the contribution of the remaining two-electron operators, which can be written as

$$H_2 = \sum_{a<b<c<d}^{2n} \sum_{G_{AB}} d_{11}(G_{AB}) \sum_{i<j}^{n} g_{ij}^{AB} \langle G^{AB}|G'^{ij}\rangle, \tag{21}$$

where G_{AB} are the subpatterns of the four elements ϕ_a, ϕ_b, ϕ_c and ϕ_d, $d_{11}(G_{AB})$ are factors associated with the subpatterns G_{AB} and take some simple values, G^{AB} are subpatterns of G without elements of ϕ_a, ϕ_b, ϕ_c and ϕ_d, and G'^{ij} are subpatterns of G' without the ith and jth pairs. Details about the pairing patterns G^{AB} and the d_{11}-values in various cases are given in the appendix. The second summation runs over three subpatterns for each set of the four elements. g_{ij}^{AB} are two-electron

quantities related to subpatterns G_{AB} and G'_{ij}. G'_{ij} are subpatterns of G' with the ith and jth pairs, i.e.

$$G'_{ij} = \begin{cases} \phi'_{2i-1} - \phi'_{2i} \\ \phi'_{2j-1} - \phi'_{2j} \end{cases} \tag{22}$$

Let us suppose

$$G_{AB} = \begin{cases} \phi_{a_1} - \phi_{a_2} \\ \phi_{b_1} - \phi_{b_2} \end{cases} \tag{23}$$

then g_{ij}^{AB} can be evaluated by

$$g_{ij}^{AB} = \sum_{k=1}^{2} \sum_{l=1}^{2} \sum_{r=2i-1}^{2i} \sum_{s=2j-1}^{2j} (g_{klrs} S_{k'r'} S_{l's'} + g_{klsr} S_{k's'} S_{l'r'}). \tag{24}$$

Here

$$g_{klrs} = \langle \phi_{a_k}(1) \phi_{b_l}(2) | g(1,2) | \phi'_r(1) \phi'_s(2) \rangle, \tag{25}$$

$$S_{kr} = \langle \phi_{a_k} | \phi'_r \rangle, \quad S_{ls} = \langle \phi_{b_l} | \phi'_s \rangle$$

and

$$k + k' = 3, \quad l + l' = 3, \quad r + r' = 4i - 1, \quad s + s' = 4j - 1.$$

In practice, the evaluation of H_2 is the most time-consuming step. Actually, all the necessary quantities $\langle G^{AB} | G'^{ij} \rangle$ for the evaluation of the overlap matrix element and H_1 are available after H_2 being obtained. With these quantities, the computational effort for evaluating the overlap matrix element and H_1 is negligible. A different feature for evaluating H_2 is that it starts from subpatterns G^{AB} and the two-electron quantities g_{ij}^{AB}. The number of the unique subpatterns G^{AB} involved in Eq. (21), denoted as $N_G(n-2)$, is given as follows (see the appendix):

$$N_G(n-2) = \frac{n(n-1)\ (4n^2 - 16n + 17)}{2}. \tag{26}$$

The evaluation of all the quantities g_{ij}^{AB} takes the major part of the CPU time if $N < 12$.

3 Efficiency of the algorithm and the decomposition of the pairing patterns

One can easily know that the expansion of the overlap matrix elements in Eq. (13) is not unique. The first summation runs over $G_A (= \phi_1 - \phi_j, j = 2,3, \ldots, 2n)$. In all these $(2n-1)$ G_A, there is one element ϕ_1 in common. Actually, we can choose any element, say, ϕ_i, as the fixed element of G_A, i.e.

$$G_A = \phi_i - \phi_j \quad (j = 1,2, \ldots, i-1, i+1, \ldots, 2n) \tag{27}$$

and G^A is subpattern of G without ϕ_i and ϕ_j. If $P_A(i) \neq j$, then G^A has one pair as $\phi_{P_A(i)} - \phi_{P_A(j)}$ i.e.

$$G^A = \begin{cases} \phi_1 & - & \phi_2 \\ & \vdots & \\ \phi_{P_A(i)} & - & \phi_{P_A(j)} \\ & \vdots & \\ \phi_{N-1} & - & \phi_N \end{cases} \tag{28}$$

Of course, there is no prescription for choosing the fixed element of G_A in the first step, and one always obtains $(2n-1)$ subpatterns G^A. The same procedure can be applied to these subpatterns, and for each subpattern G^A, there are $(2n-2)$ possible ways of pattern decomposition: thus, there are totally $(2n-1)^{(2n-2)}$ possible ways to perform the pattern decomposition of the $(2n-1)$ subpatterns G^A. The number of the unique subpatterns G^{AB} thus depends on the explicit way of decomposition. From a set of $N_G(n-2)$ subpatterns G^{AB}, there are $(2n-3)^{N_G(n-2)}$ possible ways of decomposition, and so on. Therefore, from one pattern G, there are extremely large number of ways of pattern decomposition. Figures 1 and 2 show two different ways of decomposition of a pattern of 8 elements. In Fig. 1, the pairs and the elements in each pair are rearranged according to a lexical order (see below), and after the reordering, the first element of each subpattern is chosen as the fixed one for the decomposition. Let us set a lexical order of orbitals as follows:

$$\phi_1, \phi_2, \phi_3, \phi_4, \cdots, \phi_{2n},$$

and suppose we have a pairing pattern G as below:

$$G = \begin{cases} \phi_{a_1} & - & \phi_{a_2} \\ \phi_{b_1} & - & \phi_{b_2} \\ & \vdots & \\ \phi_{z_1} & - & \phi_{z_2}. \end{cases} \tag{29}$$

We say the pairs in the pattern G are arranged in the lexical order if the following conditions are satisfied: in each pair, the first element precedes the second one, and the first elements of all pairs also follow the same order, i.e.

$$a_1 < a_2, b_1 < b_2, \ldots, z_1 < z_2, a_1 < b_1 < \cdots < z_1. \tag{30}$$

In principle, we can also define a different lexical order, and rearrange the pairs according to this order. As a comparison, Fig. 2 shows the result without such an ordering. Figures 1 and 2 indicate that with the lexical ordering, the pattern decomposition leads to a smaller number of $N_G(2)$. We have also checked all other possible ways, and found $N_G(2)$ ranging from 15 to 33. One essential problem is how to find an appropriate way of decomposition which gives the minimal number of computational operations. Generally, the lexical ordering leads to a significantly smaller number of operations than that without this ordering. This can be easily seen from Table 2. From the table, one can also see that the computation is greatly reduced. For example, for a system of 20 electrons, there are 6.5×10^8 (= 19!!) permanents of order 10, and the evaluation of each individual permanent requires 5110 operations [24]; thus, the direct summation method requires 3.3×10^{11} operations. This number is reduced by a factor of about 10^4 using the successive

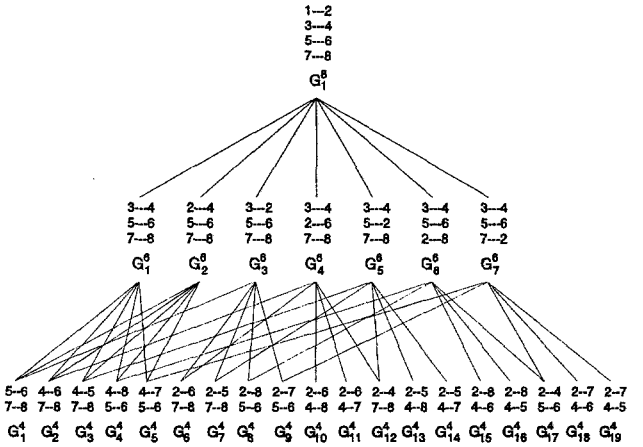

Fig. 2. The decomposition of a pattern of 8 elements without the lexical ordering

Table 2. Number of multiplications required for the evaluation of a VB overlap matrix element using different procedures of pattern decomposition

N	M_1	M_2	M	M/M_1
8	1.0×10^3	1.1×10^3	2.9×10^3	2.9
10	8.6×10^3	1.1×10^4	7.1×10^4	8.2
12	7.5×10^4	9.9×10^4	1.9×10^6	26
14	6.1×10^5	9.3×10^5	5.9×10^7	97
16	4.8×10^6	8.9×10^6	2.1×10^9	4.3×10^2
18	4.1×10^7	8.8×10^7	7.9×10^{10}	1.9×10^3
20	3.4×10^8	8.9×10^8	3.3×10^{11}	9.9×10^3

N: Number of electrons
M_1: Number of operations using the procedure with lexical ordering
M_2: Same number without ordering
M: Number of operations required for the direct summation of all permanents

expansion method shown above. For the evaluation of H_2, one starts from $n(n-1)(4n^2-16n+17)/2$ subpatterns G^{AB}. We also find that the pattern decomposition with the lexical ordering is much more optimal than that without this ordering.

Further improvement of the efficiency of the algorithm

The total number of multiplications required for the evaluation of an individual Hamiltonian matrix element can be approximately given as

$$M = \sum_{m=2}^{n-2} N_G(m) \frac{n!(2m-1)}{(m-1)!(n-m)!} + 16n^2(n-1)^2(2n-1)(2n-3). \quad (31)$$

The first term gives the number of operations for computing all the quantities $\langle G^{AB} | G^{'ij} \rangle$, and the second term gives the number for all the two-electron quanti-

ties g_{ij}^{AB}. The first term depends on the numbers of the subpatterns of various levels in the tree of the pattern decomposition. From the above discussion we know that the lexical ordering in the pattern decomposition leads to a smaller number of computational operations as compared to the one without such an ordering. In this pattern-decomposition procedure, just one lexical order is used for the rearrangement of the pairs in the subpatterns. We call this a homogeneous lexical ordering. Actually, if one follows a more sophisticated procedure for the reordering of the pairs in the subpatterns, it should be possible to find a smaller number of operations. The idea is that, instead of using a uniform lexical order, one can introduce different lexical orders for different groups of subpatterns, i.e. we divide the subpatterns G^{AB} involved in the summation of Eq. (21) into a number of groups, and introduce a unique lexical order for each of them. In the procedure of pattern decomposition, the pairs of all subpatterns with the parentage of the subpatterns G^{AB} are rearranged according to the individual lexical order of the group containing the subpatterns G^{AB}. We call this procedure an inhomogeneous lexical ordering.

In this new scheme, the lexical orders for different groups of subpatterns are shown as follows: suppose G^{AB} is a subpattern of G without four elements ϕ_a, ϕ_b, ϕ_c and ϕ_d, which are from the i, j, k and lth pairs of G, respectively. The lexical order for this subpattern is that the i, j, k and lth pairs come first, then the remaining pairs, as shown below:

$$\phi_{2i-1}\text{–}\phi_{2i}$$
$$\phi_{2j-1}\text{–}\phi_{2j}$$
$$\phi_{2k-1}\text{–}\phi_{2k}$$
$$\phi_{2l-1}\text{–}\phi_{2l}$$
$$\phi_1\text{–}\phi_2$$
$$\vdots$$
$$\phi_{N-1}\text{–}\phi_N$$

Using this scheme, the number of operations is further reduced; see Table 3. For example, the number for 20 electrons is reduced by a factor 1.7, and this factor seems to be increasing with increasing N. This new scheme has been implemented into a graph-analysis program for generating a universal file of the pattern

Table 3. Approximate estimation of the number (M) of multiplications for the evaluation of a Hamiltonian VB matrix element using different methods

N	12	14	16	18	20
M_1	2.3×10^6	1.4×10^7	1.2×10^8	1.2×10^9	1.4×10^{10}
M_2	2.3×10^6	1.3×10^7	8.8×10^7	6.9×10^8	5.9×10^9
M_3	2.2×10^6	1.0×10^7	5.9×10^7	4.3×10^8	3.3×10^9

N: Number of electrons
M_1: Number of multiplications without lexical ordering
M_2: Same number with homogeneous lexical ordering
M_3: Number with inhomogeneous lexical ordering

decomposition, while the major body of the VB program does not need any change. With the file of the pattern decomposition generated by the new procedure, the efficiency of the VB program is improved. For example, the VB calculation (one structure, without orbital optimization) of a 16-electron system requires about 1 min using the previous scheme, and the new scheme requires 35 s. The calculations were done on a IBM workstation.

4 VBSCF calculations

To test the efficiency of the new scheme, the VBSCF calculations of CH_4, C_2H_2, C_2H_4 and C_2H_6 have been performed. The optimized geometries at RHF/6–31*G* level are used. The same 6–31*G* basis sets are used for the orbital optimization. In the calculations, the nonorthogonal orbitals are expanded as linear combinations of atomic orbitals. All orbitals for the construction of VB wave functions are optimized with the super-CI method and the DIIS technique. The bottle-neck of the super-CI method is the evaluation of the matrix elements of a large number of VB structures (Brillouin states). A "global strategy" is adopted for the systematic evaluations of the super-CI matrix elements. The carbon–carbon multiple bonds can be described as either σ and π bonds or equivalent banana bonds [2]. In this paper, both types of VB wave functions are considered. The convergence threshold is set to be $\Delta C < 10^{-5}$ (ΔC is the maximal difference of the combination coefficients between two consecutive iterations). Usually, the convergence can be reached within 10–20 iterations. The calculations are done on a IBM workstation RS6000/320H. The results and the CPU time for each VBSCF iteration are shown in Table 4.

From Table 4 one can see an interesting feature of the carbon–carbon multiple bonds. For C_2H_2 and C_2H_4, the banana-type VB wave functions give slightly lower energies than that of the σ–π ones. This is in agreement with a more sophisticated SCVB calculations with larger basis sets [4]. The CPU time required for each iteration increases in a fairy reasonable manner with the increasing number of electrons. Therefore, the new group-theoretical approach presented here is quite promising.

Table 4. The Hartree–Fock energy E(HF) and the VBSCF energy E(VB) of CH_4, C_2H_2, C_2H_4 and C_2H_6 and the CPU time required for each VBSCF iteration

	CH_4	$C_2H_2(\sigma-\pi)$	C_2H_2(banana)	$C_2H_4(\sigma-\pi)$	C_2H_4(banana)	C_2H_6
E(HF)	−40.18055416	−76.79276206		−78.00554475		−79.19650556
E(VB)	−40.24281587	−76.87927942	−76.88689459	−78.10573986	−78.10835869	−79.30708079
CPU [s]	22	144		283		2300

5 Concluding remarks

In this paper, we represent a new group-theoretical approach of nonorthogonal *ab initio* VB calculations in a graphical language. One essential feature which leads to the significant reduction of the computational effort is that the intermediate quantities characterized by the subpatterns of various numbers of pairs of orbitals

are introduced. These quantities are frequently reused in the whole computation. To achieve higher efficiency of the algorithm, one needs to find an appropriate pattern decomposition which gives the minimal number of computer operations. Although this paper is dealing with the evaluation of individual VB matrix elements, one can expect that the same strategy can be applied to the systematic evaluation of all VB matrix elements. In this case, the intermediate quantities are not only frequently reused for computing the overlap and Hamiltonian VB matrix elements of an individual pair of VB wave functions, but also may be reused for computing other VB matrix elements. Such a global strategy is adopted in the super-CI method for VBSCF calculations. Similar "global strategy" was discussed by Raimondi et al. [35] and applied in SCVB calculations.

Acknowledgements. The author is very grateful to the Alexander-von-Humboldt foundation for the award of a Fellowship that makes his stay at the Chair for Theoretical Chemistry at the University Erlangen-Nürnberg possible. He is also indebted to Dr. D.L. Cooper for his hospitality during the visit to Liverpool, for giving him the opportunity to present this work at the University of Liverpool. He would like to express his thanks to Professor R. Pauncz for sending him the manuscript of his recent book. Thanks are also due to Professor Janos Ladik for his careful reading and improvement of this manuscript.

Appendix: d_{11}-values and pairing pattern of G^{AB}

Case 1. The four elements $\phi_a, \phi_b, \phi_c, \phi_d$ are from two pairs of G, say, ϕ_{2k-1}–ϕ_{2k} and ϕ_{2l-1}–ϕ_{2l}. In this case, G^{AB} is a subpattern of G without these two pairs. As there are $n(n-1)/2$ ways of choosing two pairs, one obtains $n(n-1)/2$ subpatterns G^{AB} in case 1. The d_{11}-values are shown below:

G_{AB}:	$\begin{cases}\phi_{2k-1}\text{–}\phi_{2k} & \phi_{2k-1}\text{–}\phi_{2l-1} & \phi_{2k-1}\text{–}\phi_{2l} \\ \phi_{2l-1}\text{–}\phi_{2l} & \phi_{2k}\text{–}\phi_{2l} & \phi_{2l-1}\text{–}\phi_{2k}\end{cases}$		
$d_{11}(G_{AB})$:	1	$-\frac{1}{2}$	$-\frac{1}{2}$

Case 2. The four elements are from three pairs. Thus, there are two elements, say, ϕ_a and ϕ_b, from one pair of G, i.e. $P_A(a) = b$, while the other two elements ϕ_c and ϕ_d are from other two different pairs of G, i.e. $P_A(c) \neq d$. In this case, G^{AB} is a subpattern of G consisting of a pair $\phi_{P_A(c)}$–$\phi_{P_A(d)}$. In case 2, one obtains $2n(n-1)(n-2)$ unique subpatterns G^{AB}. Finally, the d_{11}-values are given as follows:

G_{AB}:	$\begin{cases}\phi_a\text{–}\phi_b & \phi_a\text{–}\phi_c & \phi_a\text{–}\phi_d \\ \phi_c\text{–}\phi_d & \phi_b\text{–}\phi_d & \phi_c\text{–}\phi_b\end{cases}$		
$d_{11}(G_{AB})$:	$-\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$

Case 3. All the four elements are from four different pairs of G. In this case, $d_{11}(G_{AB}) = \frac{1}{4}$ for all the three subpatterns G_{AB}. G^{AB} are subpatterns of G in which the four elements of G_{AB} are taken away and the remaining four elements in the incomplete pairs form two new pairs in the following way:

G_{AB}:	$\begin{cases}\phi_a\text{–}\phi_b & \phi_a\text{–}\phi_c & \phi_a\text{–}\phi_d \\ \phi_c\text{–}\phi_d & \phi_b\text{–}\phi_d & \phi_c\text{–}\phi_b\end{cases}$

$$G^{AB}: \begin{cases} \phi_{P_A(a)}-\phi_{P_A(b)} & \phi_{P_A(a)}-\phi_{P_A(c)} & \phi_{P_A(a)}-\phi_{P_A(d)} \\ \phi_{P_A(c)}-\phi_{P_A(d)} & \phi_{P_A(b)}-\phi_{P_A(d)} & \phi_{P_A(c)}-\phi_{P_A(b)} \\ \vdots & \vdots & \vdots \end{cases}$$

One obtains $2n(n-1)(n-2)(n-3)$ subpatterns in this case. Thus, the total number of unique subpatterns G^{AB} involved in the evaluation of H_2 is given by Eq. (21).

References

1. Coulson CA (1961) Valence. Oxford Univ Press, London
2. Pauling L (1960) The nature of the chemical bond. Cornell Univ Press, Ithaca, NY
3. Sironi M, Raimondi M, Cooper DL, Gerratt J (1991) J Phys Chem 95:10617
4. Karadakov PB, Gerratt J, Cooper DL, Raimondi M (1993) J Am Chem Soc 115:6863
5. Cooper DL, Cunningham TP, Gerratt J, Karadakov PB, Raimondi M (1994) J Am Chem Soc 116:4414
6. Karadakov PB, Gerratt J, Cooper DL, Raimondi M (1994) J Am Chem Soc 116:7714
7. Sironi M, Cooper DL, Gerratt J, Raimondi M (1990) J Am Chem Soc 112:5054
8. Wu W, McWeeny R (1994) J Chem Phys 106:4826
9. Karadakov PB, Gerratt J, Cooper DL, Raimondi M (1994) J Chem Soc Faraday Trans 90:1643
10. Cooper DL, Mort KA, Allan NL, Kinchington D, McGuigan C (1993) J Am Chem Soc 115:12615
11. Gallup GA, Vance RL, Collins JR, Norbeck JM (1982) Adv Quant Chem 16:229
12. Cooper DL, Gerratt J, Raimondi M (1987) Adv Chem Phys 69:319
13. van Lenthe JH, Verbeck J, Pulay P (1991) Mol Phys 73:1159
14. Rettrup S, Thorsteinsson T, Sarma CR (1991) Int J Quant Chem 40:709
15. Cooper DL, Gerratt J, Raimondi M (1991) Chem Rev 91:929
16. Gerratt J, Cooper DL, Raimondi M (1990) In: Klein DJ, Trinajstic N (eds) Valence bond theory and chemical structure. Elsevier, New York
17. Karadakov PB, Gerratt J, Raimondi M, Cooper DL (1992) J Chem Phys 97:7635
18. Cooper DL, Gerratt J, Raimondi M, Sironi M, Thorsteinsson T (1993) Theor Chim Acta 85:261
19. Klein DJ, Junker BR (1971) J Chem Phys 54:4294; Junker BR, Klein DJ (1971) J Chem Phys 55:5532
20. McWeeny R (1988) Int J Quant Chem 34:25
21. Li X, Paldus J (1991) J Mol Struct (THEOCHEM) 229:249
22. Li X, Paldus J (1992) Int J Quant Chem 41:117
23. Poshusta RD, Kinghorn DB (1992) Int J Quant Chem 41:15
24. Li J, Wu W (1994) Theor Chim Acta 89:105
25. Li J (1995) J Math Chem 17:295
26. a) Pauncz R (1979) Spin eigenfunctions. Plenum Press, New York; (b) Pauncz R (1995) The symmetric group in quantum chemistry. CRC Press, Boca Raton, FL
27. Matsen FA (1964) Adv Quant Chem 1:50
28. Matsen FA (1992) Int J Quant Chem 41:7
29. Shavitt I (1977) Int J Quantum Chem 11s:131; Shavitt I (1978) Int J Quantum Chem 12s:5
30. Brook BR, Schaefer HF (1979) J Chem Phys 70:5092
31. Duch W, Karwowski J (1982) Int J Quantum Chem 22:783
32. (a) Minc H (1978) Permanents. Addison-Wesley, Reading, MA; (b) Ryser HJ (1965) Combinatorial mathematics, Math Assoc Amer, NY; (c) Jurkat WB, Ryser HJ (1966) J Algebra 3:1; (d) Nijenhuis A, Wilf HS (1975) Combinatorial algorithms. Academic Press, New York
33. Gerratt J (1971) Adv at Molec Phys 7:41.
34. Pyper NC, Gerratt J (1977) Proc Roy Soc Lond A355:407
35. Raimondi M, Gianinetti E (1988) J Phys Chem 92:899